Object Detection Networks on Convolutional Feature Maps
卷积特征映射上的目标检测网络

日期：2016-08-17

作者：Shaoqing Ren (/search?search_txt=Shaoqing Ren) 、Kaiming He (/search?search_txt=Kaiming He) 、Ross Girshick (/search?search_txt=Ross Girshick) 、Xiangyu Zhang (/search?search_txt=Xiangyu Zhang) 、Jian Sun (/search?search_txt=Jian Sun)

论文：http://arxiv.org/pdf/1504.06066v2.pdf (http://arxiv.org/pdf/1504.06066v2.pdf)

Shaoqing Ren, Kaiming He, Ross Girshick, Xiangyu Zhang, and Jian Sun
Shaoqing Ren, Kaiming He, Ross Girshick, Xiangyu Zhang, and Jian Sun

Abstract—Most object detectors contain two important components: a feature extractor and an object classifier. The feature extractor has rapidly evolved with significant research efforts leading to better deep convolutional architectures. The object classifier, however, has not received much attention and many recent systems (like SPPnet and Fast/Faster R-CNN) use simple multi-layer perceptrons. This paper demonstrates that carefully designing deep networks for object classification is just as important. We experiment with region-wise classifier networks that use shared, region-independent convolutional features. We call them “Networks on Convolutional feature maps” (NoCs). We discover that aside from deep feature maps, a *deep* and *convolutional* per-region classifier is of particular importance for object detection, whereas latest superior image classification models (such as ResNets and GoogLeNets) do not directly lead to good detection accuracy without using such a per-region classifier. We show by experiments that despite the effective ResNets and Faster R-CNN systems, the design of NoCs is an essential element for the 1st-place winning entries in ImageNet and MS COCO challenges 2015.

!

!

1 INTRODUCTION
1引言

Most object detectors contain two important components: a feature extractor and an object classifier. The feature extractor in traditional object detection methods is a hand-engineered module, such as HOG [1]. The classifier is often a linear SVM (possibly with a latent structure over the features) [2], a non-linear boosted classifier [3], or an additive kernel SVM [4]. Large performance improvements have been realized by training deep ConvNets [5] for object detection. R-CNN [6], one particularly successful approach, starts with a pre-trained ImageNet [7] classification network and then fine-tunes the ConvNet, end-to-end, for detection. Although the distinction between the feature extractor and the classifier becomes blurry, a logical division can still be imposed. For example, an R-CNN can be thought of as a convolutional feature extractor, ending at the last pooling layer, followed by a multi-layer perceptron (MLP) classifier. This methodology, however, appears rather different from traditional methods.

大多数对象检测器包含两个重要组件：特征提取器和对象分类器。传统物体检测方法中的特征提取器是手工设计的模块，例如HOG [1]。分类器通常是线性SVM（可能具有特征上的潜在结构）[2]，非线性增强分类器[3]或附加

核SVM [4]。通过训练深度ConvNets [5]进行物体检测，实现了大的性能改进。R-CNN [6]是一种特别成功的方法，从预先训练的ImageNet [7]分类网络开始，然后对端到端的ConvNet进行微调，以便进行检测。虽然特征提取器和分类器之间的区别变得模糊，但仍然可以施加逻辑分割。例如，R-CNN可以被认为是卷积特征提取器，在最后的池化层结束，接着是多层感知器（MLP）分类器。然而，这种方法似乎与传统方法有很大不同。

A research stream [8], [9], [10], [11] attempting to bridge the gap between traditional detectors and deep ConvNets creates a hybrid of the two: the feature extractor is "upgraded" to a pre-trained deep ConvNet, but the classifier is left as a traditional model, such as a DPM [8], [9], [10] or a boosted classifier [11]. These hybrid approaches outperform their HOGbased counterparts [2], [3], but still lag far behind RCNN, even when the hybrid model is trained endto-end [10]. Interestingly, the detection accuracy of these hybrid methods is close to that of R-CNN when • The majority of this work was done when the authors were with Microsoft Research.
试图弥合传统探测器和深度ConvNets之间差距的研究流[8]，[9]，[10]，[11]创造了两者的混合：特征提取器被"升级"为预训练的深度ConvNet但是，分类器被留作传统模型，例如DPM [8]，[9]，[10]或增强分类器[11]。这些混合方法的表现优于基于HOG的对应方[2]，[3]，但仍远远落后于RCNN，即使混合模型是端到端训练[10]。有趣的是，这些混合方法的检测准确性接近于R-CNN的时候•大部分工作是在作者与微软研究院合作时完成的。

• S. Ren is with University of Science and Technology of China. • K. He and R. Girshick are with Facebook AI Research. • X. Zhang is with Xi'an Jiaotong University. • J. Sun is with Megvii.
•S. Ren与中国科技大学合作。 •K。他和R. Girshick在Facebook AI Research工作。 •X。张在西安交通大学。 •J。Sun与Megvii合作。

using a linear SVM on the last convolutional features, without using the multiple fully-connected layers1. The SPPnet approach [12] for object detection occupies a middle ground between the hybrid models and R-CNN. SPPnet, like the hybrid models but unlike RCNN, uses convolutional layers to extract full-image features. These convolutional features are independent of region proposals and are shared by all regions, analogous to HOG features. For classification, SPPnet uses a region-wise MLP, just like R-CNN but unlike hybrid methods. SPPnet is further developed in the latest detection systems including Fast R-CNN [13] and Faster R-CNN [14], which outperform the hybrid methods.
在最后的卷积特征上使用线性SVM，而不使用多个完全连接的层1。用于物体检测的SPPnet方法[12]在混合模型和R-CNN之间占据中间地带。SPPnet与混合模型一样，但与RCNN不同，它使用卷积层来提取全图像特征。这些卷积特征独立于区域提议，并且由所有区域共享，类似于HOG特征。对于分类，SPPnet使用区域性MLP，就像R-CNN一样，但与混合方法不同。SPPnet在最新的检测系统中得到进一步发展，包括Fast R-CNN [13]和更快的R-CNN [14]，其性能优于混合方法。

From these systems [12], [13], [14], a prevalent strategy for object detection is now: use convolutional layers to extract region-independent features, followed by region-wise MLPs for classification. This strategy was, however, historically driven by pre-trained classification architectures similar to AlexNet [5] and VGG nets [15] that end with MLP classifiers.
从这些系统[12]，[13]，[14]，目前流行的物体检测策略是：使用卷积层提取与区域无关的特征，然后使用区域性MLP进行分类。然而，这种策略在历史上受到预先训练的分类架构的驱动，类似于AlexNet [5]和VGG网络[15]以MLP分类器结束。

In this paper, we provide an in-depth investigation into object detection systems from the perspective of classifiers aside from features. We focus on region-wise classifier architectures that are on top of the shared, region-independent convolutional features. We call them "Networks on Convolutional feature maps", or NoCs for short. Our study brings in

new insights for understanding the object detection systems.
在本文中，我们从除了特征之外的分类器的角度对对象检测系统进行深入研究。我们专注于区域级分类架构，这些架构位于共享的，与区域无关的卷积特征之上。我们将它们称为“卷积特征图上的网络”，或简称为NoCs。我们的研究为理解物体检测系统带来了新的见解。

Our key observation is that carefully designed region-wise classifiers improve detection accuracy over what is typically used (MLPs). We study three NoC families: MLPs of various depths, ConvNets of various depths, and ConvNets with maxout [16] for latent scale selection, where the latter two are unex
我们的关键观察是精心设计的区域分类器提高了检测精度，超过了通常使用的（MLP）。我们研究了三个NoC系列：不同深度的MLP，各种深度的ConvNets，以及带有maxout [16]的ConvNets用于潜在尺度选择，其中后两者是未来的

1. The mAP on PASCAL VOC 2007 is 45-47% [8], [9], [10], [11] for hybrid methods, and is 47% for R-CNN that just uses SVM on the last convolutional layer. Numbers are based on AlexNet [5].
1. PASCAL VOC 2007上的mAP对于混合方法是45-47％[8]，[9]，[10]，[11]，对于在最后一个卷积层上仅使用SVM的R-CNN是47％。数字基于AlexNet [5]。

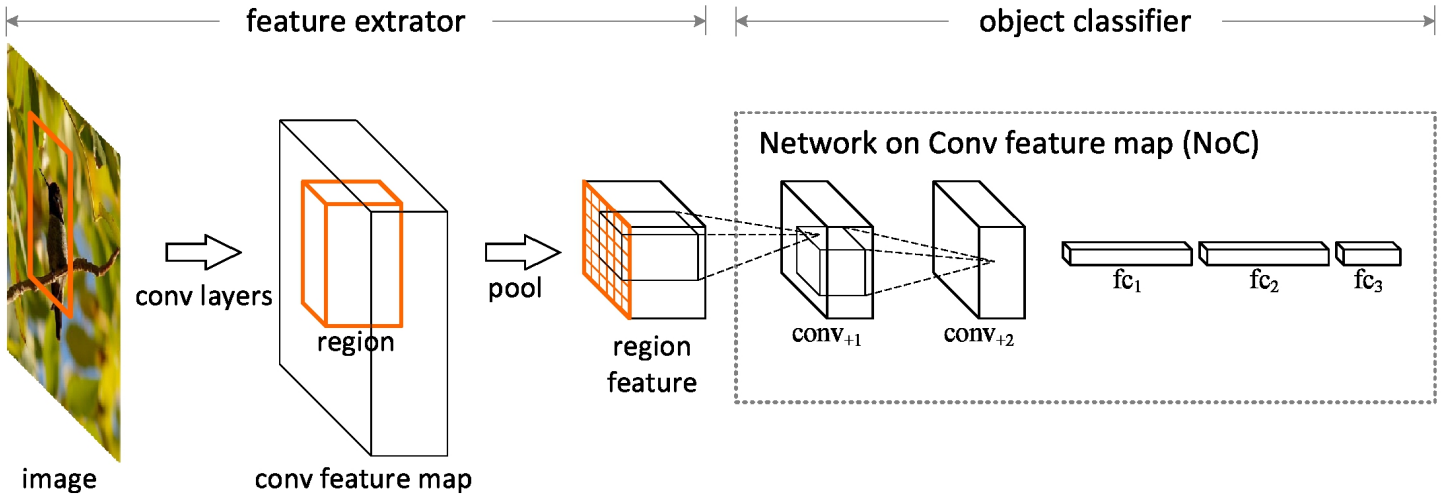


Figure 1: Overview of NoC. The convolutional feature maps are generated by the shared convolutional layers. A feature map region is extracted and RoI-pooled into a fixed-resolution feature. A new network, called a NoC, is then designed and trained on these features. In this illustration, the NoC architecture consists of two convolutional layers and three fully-connected layers.
图1：NoC概述。卷积特征图由共享卷积层生成。提取特征映射区域并将其合并为固定分辨率特征。然后，在这些功能上设计和训练称为NoC的新网络。在该图中，NoC架构由两个卷积层和三个完全连接的层组成。

plored families in previous works [12], [13], [14]. Ablation experiments suggest that: (i) a deep region-wise classifier is important for object detection accuracy, in addition to deep shared features; (ii) convolutional layers for extracting region-wise features are effective, and are complementary to the effects for extracting full-image shared features.
以前作品中的家庭[12]，[13]，[14]。消融实验表明：（i）除了深度共享特征外，深度区域分类对于物体检测准确性也很重要；（ii）用于提取区域特征的卷积层是有效的，并且与提取全图像共享特征的效果互补。

Based on these observations, we present an effective way of plugging “fully convolutional” image classifiers (such as ResNets [17] and GoogLeNets [18]) into the Faster R-CNN [14] system that was designed for the “semi-convolutional”

VGG nets [15]. We report that superior image classification backbones (e.g., ResNets and GoogLeNets) do not directly lead to better object detection accuracy, and a deep, convolutional NoC is an essential element for outstanding detection performance, in addition to Faster R-CNN and extremely deep ResNets (more details in Table 8).
基于这些观察，我们提出了一种将“完全卷积”图像分类器（如ResNets [17]和GoogLeNets [18]）插入更快的R-CNN [14]系统的有效方法，该系统是为“半卷积”而设计的。 “VGG网[15]。我们报告说，卓越的图像CLASSI网络阳离子骨架（如ResNets和GoogLeNets）并不直接带来更好的目标探测精度，和深，卷积NoC的是出色的检测性能的重要因素，除了更快的R-CNN和极深ResNets （表8中的更多细节）。

In summary, through NoC we investigate the region-wise classifiers from different aspects, which are orthogonal to the investigation of features. We believe the observations in this paper will improve the understandings of ConvNets for object detection and also boost the accuracy of prevalent detectors such as Faster R-CNN [14].
总之，通过NoC，我们从不同方面研究区域分类，这些分类与特征的研究是正交的。我们相信本文中的观察结果将提高ConvNets对物体检测的理解，并提高流行检测器的准确性，如更快的R-CNN [14]。

2 RELATED WORK
2相关工作

Traditional Object Detection. Research on object detection in general focuses on both features and classifiers. The pioneering work of Viola and Jones [19] uses simple Haar-like features and boosted classifiers on sliding windows. The pedestrian detection method in [1] proposes HOG features used with linear SVMs. The DPM method [2] develops deformable graphical models and latent SVM as a sliding-window classifier. The Selective Search paper [4] relies on spatial pyramid features [20] on dense SIFT vectors [21] and an additive kernel SVM. The Regionlet method [3] learns boosted classifiers on HOG and other features.
传统的物体检测。对象检测的研究一般侧重于特征和分类。Viola和Jones [19]的开创性工作使用了简单的Haar式功能，并在滑动窗户上提升了分类。[1]中的行人检测方法提出了与线性SVM一起使用的HOG特征。DPM方法[2]开发了可变形图形模型和潜在SVM作为滑动窗口分类器。选择性搜索论文[4]依赖于密集SIFT向量[21]和加性核SVM的空间金字塔特征[20]。Regionlet方法[3]学习了HOG和其他特征的提升分类。

ConvNet-based Object Detection. Convolutional layers can be applied to images of arbitrary size yielding proportionally-sized feature maps. In the Overfeat method [22], the fully-connected layers are used on each sliding window of the convolutional feature maps for efficient classification, localization, and detection. In the SPP-based object detection method [12], features are pooled from proposal regions [4] on convolutional feature maps, and fed into the original fully-connected layers for classifying.
基于ConvNet的对象检测。卷积层可以应用于任意大小的图像，从而产生按比例大小的特征图。在Overfeat方法[22]中，在卷积特征图的每个滑动窗口上使用完全连接的层，以进行有效的分类，定位和检测。在基于SPP的物体检测方法[12]中，在卷积特征图上从提议区域[4]汇集特征，并将其馈送到原始的完全连接的层中以进行分类。

Concurrent with this work, several papers [13], [14], [23], [24] improve on the SPPnet method, inheriting the same logical division of shared convolutional features and region-wise MLP classifiers. In Fast R-CNN [13], the shared convolutional layers are fine-tuned end-to-end through Region-of-Interest pooling layers. In Faster R-CNN [14], the shared features are also used for proposing regions and reducing the heavy proposal burdens. The “R-CNN minus R” method

在这项工作的同时，一些论文[13]，[14]，[23]，[24]改进了SPPnet方法，继承了共享卷积特征和区域MLP分类器的相同逻辑划分。在快速R-CNN [13]中，共享卷积层通过感兴趣区域池化端到端进行微调。在更快的R-CNN [14]中，共享特征也用于提出区域并减少沉重的建议负担。 “R-CNN减R”方法

[23] waives the requirement of region proposal by using pre-defined regions in the SPPnet system. In the Multi-Region method [24], the features are pooled from regions of multiple sizes to train an ensemble of models.
[23]通过在SPPnet系统中使用预先定义的区域来放弃区域提案的要求。在多区域方法[24]中，从多个大小的区域合并特征以训练模型的集合。

Despite the improvements, these systems [12], [13], [14], [23], [24] all use MLPs as region-wise classifiers. This logical division naturally applies to a series of networks, such as AlexNet [5], Zeiler and Fergus's (ZF) net [25], OverFeat [22], and VGG nets [15], which all have multiple fine-tunable fc layers. But this is not the case for fully convolutional classification networks, e.g., ResNet [17] and GoogleNet [18], that have no hidden fully-connected (fc) layers. We show that it is nontrivial for Fast/Faster R-CNN to achieve good accuracy using this type of networks.
尽管有所改进，但这些系统[12]，[13]，[14]，[23]，[24]都使用MLP作为区域分类。这种逻辑划分自然适用于一系列网络，如AlexNet [5]，Zeiler和Fergus（ZF）网[25]，OverFeat [22]和VGG网[15]，它们都具有多个可调谐fc层。但对于没有隐藏的完全连接（fc）层的完全卷积分类网络，例如ResNet [17]和GoogleNet [18]，情况并非如此。我们表明，快速/快速的R-CNN使用这种类型的网络实现良好的准确性是非常重要的。

3 ABLATION EXPERIMENTS
3次烧伤实验

Firstly we present carefully designed ablation experiments on the PASCAL VOC dataset [26]. We note that experiments in this section are mainly designed based on the SPPnet system. Particularly, in this section we consider the following settings: (i) the shared feature maps are frozen (which are fine-tunable with Fast R-CNN [13]) so we can focus on the classifiers; (ii) the proposals are pre-computed from Selective Search [4] (which can be replaced by a Region Proposal Network (RPN) [14]), and (iii) the training step ends with a post-hoc SVM (in contrast to the end-to-end softmax classifier in Fast R-CNN [13]). We remark that observations in this section are in general valid when these restricted conditions are relaxed or removed [13], [14], as shown in the next section with Faster R-CNN [14].
首先，我们在PASCAL VOC数据集上提出了精心设计的消融实验[26]。我们注意到本节中的实验主要是基于SPPnet系统设计的。特别是，在本节中我们考虑以下设置：（i）共享特征映射被冻结（可通过Fast R-CNN [13]进行调整），因此我们可以专注于分类器;（ii）提议是从选择性搜索[4]（可以由区域提案网络（RPN）[14]取代）预先计算出来的，以及（iii）训练步骤以事后SVM结束（相比之下）在快速R-CNN [13]中的端到端softmax分类器。我们注意到，当放宽或去除这些限制条件时，本节中的观察结果通常是有效的[13]，[14]，如下一节中更快的R-CNN [14]所示。

Experimental Settings We experiment on the PASCAL VOC 2007 set [26]. This dataset covers 20 object categories, and performance is measured by mAP on the test set of 5k images. We investigate two sets of training images: (i) the original trainval set of 5k images in VOC 2007, and (ii) an augmented set of 16k images that consists of VOC 2007 trainval images and VOC 2012 trainval images, following [27].
实验设置我们在PASCAL VOC 2007套装上进行实验[26]。该数据集涵盖20个对象类别，并且通过mAP在5k图像的测试集上测量性能。我们调查了两组训练图像：（i）VOC 2007中的5k图像的原始火车组，以及（ii）增强的16k

图像集，包括VOC 2007火车图像和VOC 2012火车图像，[27]。

As a common practice [6], [12], we adopt deep CNNs pre-trained on the 1000-class ImageNet dataset
作为一种常见做法[6]，[12]，我们采用了在1000级ImageNet数据集上预训练的深度CNN

[7] as feature extractors. In this section we investigate Zeiler and Fergus's (ZF) model [25] and VGG models [15]. The ZF model has five convolutional (conv) layers and three fully-connected (fc) layers. We use a ZF model released by [12]2. The VGG-16/19 models have 13/16 conv layers and three fc layers, released by [15]3.
[7]作为特征提取器。在本节中，我们将研究Zeiler和Fergus（ZF）模型[25]和VGG模型[15]。ZF模型具有五个卷积（conv）层和三个完全连接（fc）层。我们使用[12] 2发布的ZF模型。 VGG-16/19型号具有13/16转换层和3个fc层，由[15] 3发布。

Outline of Method We apply the conv layers of a pre-trained model to compute the convolutional feature map of the entire image. As in [12], we extract feature maps from multiple image scales. In this section these pre-trained conv layers are frozen and not further tuned as in [12], so we can focus on the effects of NoCs.
方法概述我们应用预训练模型的conv层来计算整个图像的卷积特征图。如[12]所示，我们从多个图像尺度中提取特征图。在本节中，这些预先训练好的转换层被冻结，不再像[12]中那样进一步调整，因此我们可以关注NoC的影响。

We extract ~2,000 region proposals by Selective Search [4]. We pool region-wise features from the shared conv feature maps using Region-of-Interest (RoI) pooling [13], [12]. RoI pooling produces a fixedresolution ($m \times m$) feature map for each region, in place of the last pooling layer in the pre-trained model (6×6 for ZF net and 7×7 for VGG-16/19). The pooled feature map regions can be thought of as tiny multichannel images (see Fig. 1).
我们通过选择性搜索[4]提取了~2,000个地区的提案。我们使用感兴趣区域（RoI）池[13]，[12]从共享转换特征映射中汇集区域特征。RoI池为每个区域生成固定分辨率（$m \times m$）特征图，代替预训练模型中的最后一个池层（ZFnet为6×6，VGG-16/19为7×7）。汇集的特征映射区域可以被认为是微小的多通道图像（参见图1）。

We consider these $m \times m$ -sized feature maps as a new data source and design various NoC architectures to classify these data. The NoC structures have multiple layers, and the last layer is an (n+1)way classifier for n object categories plus background, implemented by an (n+1)-d fc layer followed by softmax. Each NoC is trained by backpropagation and stochastic gradient descent (SGD). After network training, we use the second-to-last fc layer in the NoC to extract features from regions, and train a linear SVM classifier for each category using these features, for a fair comparison with [6], [12]. The implementation details follow those in [12].
我们将这些$m \times m$大小的特征映射视为新的数据源，并设计各种NoC架构来对这些数据进行分类。NoC结构具有多个层，最后一层是n个对象类别加背景的（n + 1）方式分类器，由（n + 1）-d fc层后跟softmax实现。每个NoC都通过反向传播和随机梯度下降（SGD）进行训练。在网络训练之后，我们使用NoC中倒数第二个fc层从区域中提取特征，并使用这些特征为每个类别训练线性SVM分类器，以便与[6]，[12]进行公平比较。实现细节遵循[12]中的细节。

For inference, the RoI-pooled features are fed into
为了推断，RoI-pooled特征被输入

2. https://github.com/ShaoqingRen/SPP net/

2. https://github.com/ShaoqingRen/SPP net /

3. www.robots.ox.ac.uk/~vgg/research/very deep/ the NoC till the second-to-last fc layer. The SVM classifier is then used to score each region, followed by non-maximum suppression [6].

3. www.robots.ox.ac.uk/~vgg/research/very deep / NoC直到倒数第二个fc层。然后使用SVM分类器对每个区域进行评分，然后进行非最大抑制[6]。

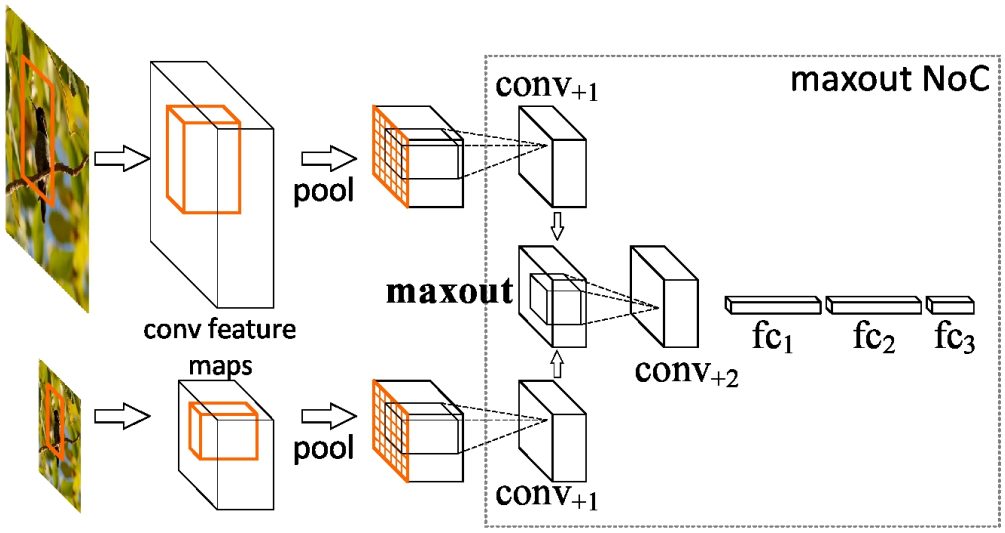


Figure 2: A maxout NoC of "c256-mo-c256-f4096f4096-f21". The features are RoI-pooled from two feature maps computed at two scales. In this figure, maxout is used after conv+1.

图2："c256-mo-c256-f4096f4096-f21"的最大NoC。这些特征是通过两个尺度计算的两个特征图进行RoI-pooled。在这个图中，在conv + 1之后使用maxout。

Next we design and investigate various NoC architectures as classifiers on the RoI-pooled features.

接下来，我们设计和研究各种NoC架构作为RoI-pooled功能的分类。

3.1 Using MLP as NoC

3.1使用MLP作为NoC

A simple design of NoC is to use fc layers only, known as a multi-layer perceptron (MLP) [28]. We investigate using 2 to 4 fc layers. The last fc layer is always (n+1)d with softmax, and the other fc layers are 4,096-d (with ReLU [29]). For example, we denote the NoC structure with 3 fc layers as "f4096-f4096-f21" where 21 is for the VOC categories (plus background).

NoC的简单设计是仅使用fc层，称为多层感知器（MLP）[28]。我们研究使用2到4个fc层。最后一个fc层总是（n + 1）d，其中softmax，其他fc层是4,096-d（使用ReLU [29]）。例如，我们将具有3个fc层的NoC结构表示为"f4096-f4096-f21"，其中21表示VOC类别（加上背景）。

Table 1 shows the results of using MLP as NoC. Here we randomly initialize the weights by Gaussian distributions. The accuracy of NoC with 2 to 4 fc layers increases with the depth. Compared with the SVM classifier trained on the RoI features ("SVM on RoI", equivalent to a 1-fc structure), the 4-fc NoC as a classifier on the same features has 7.8% higher mAP. Note that in this comparison the NoC classifiers have no pre-training (randomly initialized). The gain is solely because that MLPs are better classifiers than single-layer SVMs. In the special case of 3 fc layers, the NoC becomes a structure similar to the regionwise classifiers popularly used in SPPnet [12] and Fast/Faster R-CNN [13], [14].

表1显示了使用MLP作为NoC的结果。在这里，我们通过高斯分布随机初始化权重。具有2至4个fc层的NoC的精度随深度而增加。与在RoI特征（“SVM on RoI”，相当于1-fc结构）上训练的SVM分类器相比，4-fc NoC作为相同特征的分类器具有高7.8％的mAP。请注意，在此比较中，NoC分类器没有预训练（随机初始化）。增益完全是因为MLP是比单层SVM更好的分类。在3 fc层的特殊情况下，NoC变成了类似于SPPnet [12]和Fast / Faster R-CNN [13]，[14]中常用的区域分类器的结构。

3.2 Using ConvNet as NoC

3.2使用ConvNet作为NoC

In recent detection systems [12], [13], [14], [23], [24], conv layers in the pre-trained models are thought of as region-independent feature extractors, and thus are shared on the entire image without being aware of the regions that are of interest. Although this is a computationally efficient solution, it misses the opportunities of using conv layers to learn region-aware features that are fit to the regions of interest (instead of full images). We investigate this issue from the NoC perspective, where the NoC classifiers may have their own conv layers.

在最近的检测系统[12]，[13]，[14]，[23]，[24]中，预训练模型中的转换层被认为是与区域无关的特征提取器，因此在整个图像上共享没有意识到感兴趣的区域。虽然这是一种计算上有效的解决方案，但它错过了使用转换层来学习与感兴趣区域（而不是完整图像）相关的区域感知特征的机会。我们从NoC的角度研究这个问题，其中NoC分类器可能有自己的转换层。

method	architecture	VOC 07
SVM on RoI	f21	45.8
2fc NoC	f4096-f21	49.0
3fc NoC	f4096-f4096-f21	53.1
4fc NoC	f4096-f4096-f4096-f21	**53.6**

Table 1: Detection mAP (%) of **NoC as MLP** for PASCAL VOC 07 using a ZF net. The training set is PASCAL VOC 07 trainval. The NoCs are randomly initialized. No bbox regression is used.

method	architecture	VOC 07	07+12
3fc NoC	f4096-f4096-f21	53.1	56.5
1conv3fc NoC	c256-f4096-f4096-f21	**53.3**	58.5
2conv3fc NoC	c256-c256-f4096-f4096-f21	51.4	**58.9**
3conv3fc NoC	c256-c256-c256-f4096-f4096-f21	51.3	58.8

Table 2: Detection mAP (%) of **NoC as ConvNet** for PASCAL VOC 07 using a ZF net. The training sets are PASCAL VOC 07 trainval and 07+12 trainval respectively. The NoCs are randomly initialized. No bbox regression is used.

method	architecture	VOC 07+12
2conv3fc NoC	c256-c256-f4096-f4096-f21	58.9
mo input	**mo**-c256-c256-f4096-f4096-f21	60.1
mo conv	c256-**mo**-c256-f4096-f4096-f21	**60.7**

mo conv_{+1}	c256-**mo**-c256-f4096-f4096-f21	60.7
mo fc_1	c256-c256-f4096-**mo**-f4096-f21	60.3
mo output	c256-c256-f4096-f4096-f21-**mo**	60.1

Table 3: Detection mAP (%) of **maxout NoC** for PASCAL VOC 07 using a ZF net. The training set is 07+12 trainval. The NoCs are randomly initialized. No bbox regression is used.

We investigate using 1 to 3 additional conv layers (with ReLU) in a NoC. We use 256 conv filters for the ZF net and 512 for the VGG net. The conv filters have a spatial size of 3×3 and a padding of 1, so the $m \times m$ spatial resolution is unchanged. After the last additional conv layer, we apply three fc layers as in the above MLP case. For example, we denote a NoC with 2 conv layers as "c256-c256-f4096-f4096-f21".
我们研究在NoC中使用1到3个额外的转换层（使用ReLU）。我们对ZF网使用256个转换滤波器，为VGG网使用512个转换滤波器。转换器的空间大小为3×3，填充为1，因此$m \times m$的空间分辨率不变。在最后一个额外的转换层之后，我们应用三个fc层，如上面的MLP情况。例如，我们将具有2个转换层的NoC表示为"c256-c256-f4096-f4096-f21"。

In Table 2 we compare the cases of no conv layer (3-layer MLP) and using 1 to 3 additional conv layers. Here we still randomly initialize all NoC layers. When using VOC 07 trainval for training, the mAP is nearly unchanged when using 1 additional conv layer, but drops when using more conv layers. We observe that the degradation is a result of overfitting. The VOC 07 trainval set is too small to train deeper models. However, NoCs with conv layers show improvements when trained on the VOC 07+12 trainval set (Table 2). For this training set, the 3fc NoC baseline is lifted to 56.5% mAP. The advanced 2conv3fc NoC improves over this baseline to 58.9%. This justifies the effects of the additional conv layers. Table 2 also shows that the mAP gets saturated when using 3 additional conv layers.
在表2中，我们比较了无转换层（3层MLP）和使用1到3个额外转换层的情况。在这里，我们仍然随机初始化所有NoC图层。当使用VOC 07 trainval进行训练时，使用1个额外的转换层时，mAP几乎没有变化，但是当使用更多转换层时，mAP会下降。我们观察到降解是过度配置的结果。VOC 07 trainval set太小，无法训练更深的模型。然而，在VOC 07 + 12列车组训练时，具有转换层的NoC显示出改进（表2）。对于此训练集，3fc NoC基线提升至56.5％mAP。先进的2conv3fc NoC比这个基线提高了58.9％。这可以解决附加转换层的影响。表2还显示，当使用3个额外的conv层时，mAP变得饱和。

Using a ConvNet as a NoC is not only effective for the ZF and VGG nets. In fact, as we show in the next section (Table 8), this design is of central importance for Faster R-CNN using ResNets [17] and other fully convolutional pre-trained architectures.
使用ConvNet作为NoC不仅对ZF和VGG网络有效。事实上，正如我们在下一节（表8）中所示，这种设计对于使用ResNets [17]和其他完全卷积预训练架构的快速R-CNN至关重要。

method	model	init.	VOC 07	07+12
SVM on RoI	ZF	-	45.8	47.7
3fc NoC	ZF	random	53.1	56.5
		pre-trained	**55.8**	**58.0**
maxout 2conv3fc NoC	ZF	random	54.7	60.7
		pre-trained	**57.7**	**62.9**

		pre-trained	57.7	62.9
maxout 2conv3fc NoC	VGG-16	random	59.4	65.0
		pre-trained	**63.3**	**68.8**

Table 4: Detection mAP (%) of NoC for PASCAL VOC 07 using ZF/VGG-16 nets with different initialization. The training sets are PASCAL VOC 07 trainval and PASCAL VOC 07+12 trainval respectively. No bounding box regression is used.
表4：使用具有不同初始化的ZF / VGG-16网络检测PASCAL VOC 07的NoC的mAP（％）。训练集分别是PASCAL VOC 07 trainval和PASCAL VOC 07 + 12 trainval。不使用边界框回归。

3.3 Maxout for Scale Selection
3.3比例选择的最大值

Our convolutional feature maps are extracted from multiple discrete scales, known as a feature pyramid [2]. In the above, a region feature is pooled from a single scale selected from the pyramid following [12]. Next, we incorporate a local competition operation (maxout) [16] into NoCs to improve scale selection from the feature pyramid.
我们的卷积特征图是从多个离散尺度中提取的，称为特征金字塔[2]。在上文中，从[12]之后的金字塔中选择的单个尺度合并区域特征。接下来，我们将一个本地竞争操作（maxout）[16]纳入NoCs，以改善特征金字塔的比例选择。

To improve scale invariance, for each proposal region we select two adjacent scales in the feature pyramid. Two fixed-resolution ($m \times m$) features are RoI-pooled, and the NoC model has two data sources. Maxout [16] (element-wise max) is a widely considered operation for merging two or multiple competing sources. We investigate NoCs with maxout used after different layers. For example, the NoC model of "c256mo-c256-f4096-f4096-f21" is illustrated in Fig. 2. When the maxout operation is used, the two feature maps (for the two scales) are merged into a single feature of the same dimensionality using element-wise max. There are two pathways before the maxout, and we let the corresponding layers in both pathways share their weights. Thus the total number of weights is unchanged when using maxout.
为了改善比例不变性，我们在每个提议区域中选择要素金字塔中的两个相邻比例。两个固定分辨率（$m \times m$ ）功能是RoI-pooled，NoC模型有两个数据源。Maxout [16]（逐元素最大）是一种广泛考虑的用于合并两个或多个竞争源的操作。我们研究了在不同层之后使用maxout的NoC。例如，"c256mo-c256-f4096-f4096-f21"的NoC模型如图2所示。使用maxout操作时，使用element-wise max将两个要素图（对于两个比例）合并为具有相同维度的单个要素。在maxout之前有两条路径，我们让两条路径中的相应层共享它们的权重。因此，使用maxout时，权重总数不变。

Table 3 shows the mAP of the four variants of maxout NoCs. Their mAP is higher than that of the non-maxout counterpart, by up to 1.8% mAP. We note that the gains are observed for all variants of using maxout, while the differences among these variants are marginal.
表3显示了maxout NoC的四种变体的mAP。他们的mAP高于非maxout对应的mAP，高达1.8％mAP。我们注意到，使用maxout的所有变体都观察到了增益，而这些变体之间的差异是微不足道的。

3.4 Fine-tuning NoC
3.4微调NoC

In the above, all NoC architectures are initialized randomly. Whenever possible, we can still transfer weights from a pre-

trained architecture and fine-tune the NoCs. The comparison of random initialization vs. fine-tuning provides new insights into the impacts of the well established fine-tuning strategy [6].
在上文中，所有NoC架构都是随机初始化的。只要有可能，我们仍然可以从预先训练的架构中传输权重并调整NoC。随机初始化与微调的比较为完善的微调策略的影响提供了新的见解[6]。

For the fine-tuning version, we initialize the two 4096-d layers by the two corresponding fc layers in the pre-trained model. As such, the fine-tuned 3-fc NoC becomes equivalent to the SPPnet object detection system [12]. For the cases of additional conv layers, each conv layer is initialized to the identity mapping, and thus the initial network state is equivalent to the pre-trained 3fc structure. We compare the results of an SVM on RoI, randomly initialized NoC, and finetuned NoC initialized in the above way. Table 4 shows the cases of two NoCs.
对于微调版本，我们通过预训练模型中的两个相应fc层初始化两个4096-d层。因此，经过调整的3-fc NoC等同于SPPnet物体检测系统[12]。对于附加转换层的情况，每个转换层被初始化为标识映射，因此初始网络状态等同于预训练的3fc结构。我们比较了SVI上的SVM，随机初始化的NoC和以上述方式初始化的fi netuned NoC的结果。表4显示了两个NoC的情况。

	NoC	depth (feature)	depth (classifier)	depth (total)	mAP (%)
VGG-16	3fc	13	3	16	64.6
VGG-19	3fc	16	3	19	65.1
VGG-16	2conv3fc	13	5	18	**66.1**
VGG-16	maxout 2conv3fc	13	5	18	**68.8**

Table 5: Detection results for PASCAL VOC 07 using VGG nets. The training set is PASCAL VOC 07+12 trainval. The NoC is the fine-tuned version (Sec. 3.4). No bounding box regression is used.
表5：使用VGG网的PASCAL VOC 07的检测结果。训练集是PASCAL VOC 07 + 12 trainval。 NoC是经过精心调整的版本（第3.4节）。不使用边界框回归。

Unsurprisingly, the fine-tuned models boost the results. However, it is less expected to see that the randomly initialized NoCs produce excellent results. Compared with the SVM counterpart using the same RoI-pooled features (47.7%, Table 4), the randomly initialized NoC (60.7%) showcases an improvement of 13.0%, whereas the fine-tuned counterpart (62.9%) has an extra 2.2% gain. This indicates that the fine-tuning procedure, for the classifier, can obtain a majority of accuracy via training a deep network on the detection data.
不出所料，经过精心调整的模型可以提升效果。然而，不太可能看到随机初始化的NoC产生优异的结果。与使用相同RoI-pooled特征的SVM对应物（47.7％，表4）相比，随机初始化的NoC（60.7％）显示出13.0％的改善，而经过调整的对应物（62.9％）则增加了2.2％。获得。这表明，对于分类器，微调程序可以通过在检测数据上训练深度网络来获得大部分精度。

3.5 Deep Features vs. Deep Classifiers
3.5深度功能与深度分类器

We further show by experiments that a deep classifier has complementary effects to deep features. Table 5 shows the NoC results using the VGG models [15]. The mAP of the baseline 3fc NoC is 64.6% with VGG
我们通过实验进一步证明，深度分类器对深层特征具有互补效应。表5显示了使用VGG模型的NoC结果[15]。使用

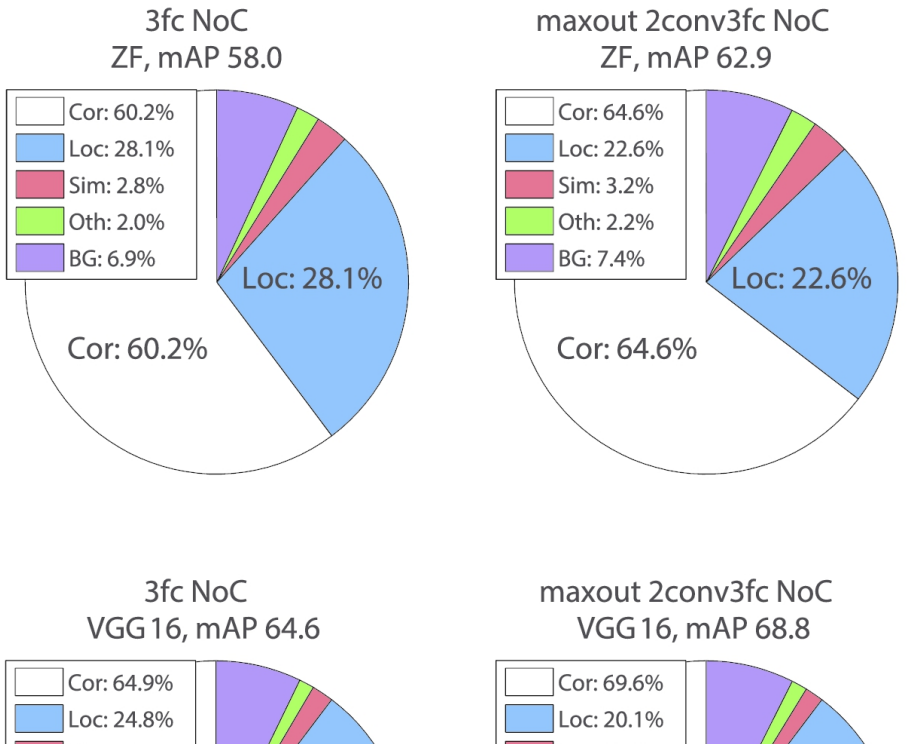
VGG时，基线3fc NoC的mAP为64.6％

16. With the network replaced by the deeper VGG-19, the depth of shared features is increased by 3, and the mAP is increased by 0.5% to 65.1%. On the other hand, when the depth of region-aware classifier is increased (but still using the VGG-16 features), the mAP is increased by 1.5% to 66.1%. This means that for exploiting very deep networks, the depth of features and the depth of classifiers are both important.
16.随着更深层次的VGG-19取代网络，共享功能的深度增加了3，mAP增加了0.5％，达到65.1％。另一方面，当区域感知分类器的深度增加（但仍然使用VGG-16特征）时，mAP增加1.5％至66.1％。这意味着，为了利用非常深的网络，特征的深度和分类的深度都很重要。

3.6 Error Analysis
3.6错误分析

Our best NoC using VGG-16 has 68.8% mAP (Table 5). To separately investigate the gains that are caused by features (stronger pre-trained nets) and classifiers (stronger NoCs), in Fig. 3 we analyze the errors of using two sets of pre-trained features (ZF vs. VGG16) and two NoCs (3fc vs. maxout 2conv3fc). We use the diagnosis tool of [30].
使用VGG-16的最佳NoC含有68.8％mAP（表5）。为了分别研究由特征（更强的预训练网）和分类器（更强的NoC）引起的增益，在图3中我们分析了使用两组预训练特征（ZF与VGG16）和两个NoC的误差（3fc vs. maxout 2conv3fc）。我们使用[30]的诊断工具。

The errors can be roughly decomposed into two parts: localization error and recognition error. Localization error (“Loc”) is defined [30] as the false positives that are correctly categorized but have no sufficient overlapping with ground truth. Recognition error involves confusions with a similar category (“Sim”), confusions with a dissimilar category (“Oth”), and confusions with background (“BG”).
误差可以粗略地分解为两部分：定位误差和识别误差。定位误差（“Loc”）被定义为[30]，因为误报被正确分类但与地面实况没有足够的重叠。识别错误涉及与类似类别（“Sim”）的混淆，与不同类别（“Oth”）的混淆以及与背景的混淆（“BG”）。



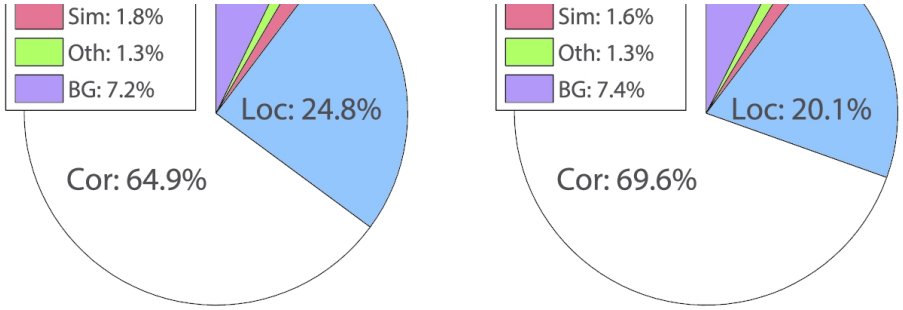


Figure 3: Distribution of top-ranked True Positives (TP) and False Positives (FP), generated by the published diagnosis code of [30]. The types of positive predictions are categorized [30] as Cor (correct), Loc (false due to poor localization), Sim (confusion with a similar category), Oth (confusion with a dissimilar category), BG (fired on background). The total number of samples in each disk is the same and equal to the total number of ground-truth labels [30]. More explanations are in the main text.
图3：由[30]公布的诊断代码生成的排名靠前的真阳性（TP）和假阳性（FP）的分布。阳性预测的类型被分类为[30]为Cor（正确），Loc（由于定位不良而假），Sim（与类似类别混淆），Oth（与不同类别混淆），BG（背景为红色）。每个磁盘中的样本总数相同，并且等于地面实况标签的总数[30]。正文中有更多解释。

Fig. 3 shows that VGG-16 in general has lower recognition error than the ZF net, when using the same classifiers (e.g., 1.6%+1.3%+7.4% vs. 3.2%+2.2%+7.4%). This suggests that the region-independent features perform more prominent for recognizing object categories. On the other hand, when using a stronger NoC (maxout 2conv3fc), the localization error is substantially reduced compared with the 3fc baseline (22.6% vs. 28.1% with ZF, and 20.1% vs. 24.8% with VGG-16). This suggests that the NoCs mainly account for localizing objects. This can be explained by the fact that localization-sensitive information is only extracted after RoI pooling and is used by NoCs.
图3显示当使用相同的分类时，VGG-16通常具有比ZF网更低的识别误差（例如，1.6％+ 1.3％+ 7.4％对3.2％+ 2.2％+ 7.4％）。这表明与区域无关的特征在识别对象类别方面表现得更为突出。另一方面，当使用更强的NoC（maxout 2conv3fc）时，与3fc基线相比，定位误差显着降低（ZF为22.6％对28.1％，VGG-16为20.1％对24.8％）。这表明NoCs主要用于本地化对象。这可以通过以下事实来解释：仅在RoI池化之后提取并且由NoC使用的本地化敏感信息。

3.7 Comparisons of Results
3.7结果比较

In Table 6 and Table 7, we provide system comparisons with recent state-of-the-art results, including RCNN [6], SPPnet [12], and the latest Fast/Faster RCNN [13], [14] that are contemporary to this work. We note that all methods in Table 6 and Table 7 are based on Selective Search (SS) proposals [4] (∼2,000 regions per image), except for Faster R-CNN [14] that uses learned proposals.
在表6和表7中，我们提供了系统与最新的最新结果的比较，包括RCNN [6]，SPPnet [12]和最新的Fast / Faster RCNN [13]，[14]这项工作。我们注意到表6和表7中的所有方法都基于选择性搜索（SS）提议[4]（每个图像约2,000个区域），除了使用学习提议的更快的R-CNN [14]。

Our method achieves 71.6% mAP on the PASCAL VOC 2007 test set. This accuracy is higher than Fast R-CNN [13] that also uses SS proposals, and lower than Faster R-CNN [14] that uses learned proposals.
我们的方法在PASCAL VOC 2007测试装置上实现了71.6％的mAP。这种准确性高于使用SS提议的快速R-CNN

[13]，并且低于使用学习提议的更快的R-CNN [14]。

method	training data	mAP (%)
R-CNN [6]	07	62.2
R-CNN [6] + bb	07	66.0
SPPnet [12]	07	60.4
SPPnet [12]	07+12	64.6
Fast R-CNN [13]	07+12	70.0
Faster R-CNN [14]	07+12	**73.2**
NoC [ours]	07+12	68.8
NoC [ours] + bb	07+12	**71.6**

Table 6: Detection results for the PASCAL VOC 2007 test set using the VGG-16 model [15]. Here "bb" denotes post-hoc bounding box regression [6].
表6：使用VGG-16模型的PASCAL VOC 2007测试装置的检测结果[15]。这里"bb"表示事后边界框回归[6]。

method	training data	mAP (%)
R-CNN [6]	12	59.2
R-CNN [6] + bb	12	62.4
Fast R-CNN [13]	07++12	68.4
Faster R-CNN [14]	07++12	**70.4**
NoC [ours]	07+12	67.6
NoC [ours] + bb	07+12	**68.8**

Table 7: Detection results for the PASCAL VOC 2012 test set using the VGG-16 model [15]. Here "bb" denotes post-hoc bounding box regression [6].
表7：使用VGG-16模型的PASCAL VOC 2012测试装置的检测结果[15]。这里"bb"表示事后边界框回归[6]。

Nevertheless, Fast/Faster R-CNN [13], [14] essentially applies a 3-fc NoC structure as the region-wise classifier, and thus the effect of NoCs is orthogonal to theirs. This effect is particularly prominent using the ResNets [17] as we show in the next section.
然而，快速/更快的R-CNN [13]，[14]基本上采用3-fc NoC结构作为区域分类器，因此NoC的效果与它们正交。使用ResNets [17]，这一效果尤为突出，如下一节所示。

3.8 Summary of Observations
3.8观察摘要

The following key observations can be concluded from the above subsections:
从以上小节可以得出以下主要观察结果：

(i) A deeper region-wise classifier is useful and is in general orthogonal to deeper feature maps.
（i）更深入的区域分类器是有用的，并且通常与更深的特征映射正交。

(ii) A convolutional region-wise classifier is more effective than an MLP-based region-wise classifier.

（ii）卷积区域分类法比基于MLP的区域分类法更有效。

These observations are strongly supported by the experimental results on the more challenging MS COCO dataset (Table 8), as we introduced in the next section.
正如我们在下一节中介绍的那样，对更具挑战性的MS COCO数据集（表8）的实验结果强烈支持这些观察结果。

4 NoC for Faster R-CNN with ResNet

The Fast/Faster R-CNN systems [13], [14] have shown competitive accuracy and speed using VGG nets. For networks similar to ZF and VGG-16, Fast/Faster RCNN are naturally applicable and their region-wise classifiers are 3fc NoCs. However, for “fully convolutional” models such as GoogleNets [18] and ResNets [17], there are no hidden fc layers for building regionwise classifiers. We demonstrate that the NoC design is an essential factor for Faster R-CNN [14] to achieve superior results using ResNets.
快速/快速的R-CNN系统[13]，[14]使用VGG网络展示了具有竞争力的准确性和速度。对于类似于ZF和VGG-16的网络，快速/更快的RCNN自然适用，其区域分类器是3fc NoC。然而，对于像GoogleNets [18]和ResNets [17]这样的“完全卷积”模型，没有用于构建区域分类器的隐藏fc层。我们证明了NoC设计是使用ResNets获得更好结果的更快R-CNN [14]的重要因素。

Experimental Settings In this section we experiment on the more challenging MS COCO dataset [31] with 80 categories. We train the models on the 80k train set, and evaluate on the 40k val set. We evaluate both COCO-style AP (@ IoU ∈ [0.5, 0.95]) as well as AP@0.5 and AP@0.75. We adopt the same hyper-parameters as in [17] for training Faster R-CNN on MS COCO.
实验设置在本节中，我们将对具有80个类别的更具挑战性的MS COCO数据集[31]进行实验。我们在80k列车上训练模型，并在40k val组上进行评估。我们评估COCO风格的AP（@IoU∈[0.5,0.95]）以及AP@0.5和AP@0.75。我们采用与[17]中相同的超参数来训练MS COCO上更快的R-CNN。

We compare network architectures of VGG-16 [15], GoogleNet [18], and ResNet-101 [17]. The VGG-16 has center crop top-1 error of 28.5% on the ImageNet classification val set. Regarding GoogleNet, we train the BN-Inception model [32] on ImageNet classification. Our reproduced GoogleNet has center crop top-1 error of 26.4%, close to that reported in [32] (25.2%). The 101-layer ResNet is released by the authors of [17], with center crop top-1 error of 23.6%. Both GoogleNet and ResNet have no hidden fc layer, and instead end with global average pooling and a 1000-d classifier.
Unlike the above section that is based on the SPPnet framework, in this section we use the more advanced Faster R-CNN [14] detector. The main differences are: (i) the entire networks including the features are finetuned end-to-end [13]; (ii) the proposals are learned by a RPN [14] with features shared; (iii) instead of post-hoc SVM, a softmax classifier and a jointly learned bounding box regressor [13] are learned endto-end. Nevertheless, these differences do not affect the design of the NoCs.
我们比较了VGG-16 [15]，GoogleNet [18]和ResNet-101 [17]的网络架构。VGG-16在ImageNet分类设定中的中心作物前1错误为28.5％。关于GoogleNet，我们在ImageNet分类上训练BN-Inception模型[32]。我们转载的GoogleNet中心作物前1错误率为26.4％，接近[32]（25.2％）报告的误差。101层ResNet由[17]的作者发布，中心作物前1错误为23.6％。GoogleNet和ResNet都没有隐藏的fc层，而是以全球平均合并和1000-d分类结束。与上面基于SPPnet框架的部分不同，在本节中我们使用更先进的更快的R-CNN [14]探测器。主要区别在于：（i）包括特征在内的整个网络是端到端的网络[13]；（ii）提案由RPN [14]学习，共享功能；（iii）代替临时SVM，softmax分类器和联合

学习的边界框回归器[13]是学习的端到端。然而，这些差异不会影响NoC的设计。

Experimental Results Table 8 shows the results on MS COCO val. We discuss by diving the results into 3 cases as following.
实验结果表8显示了MS COCO val的结果。我们通过将结果潜入3个案例进行讨论如下。

Na¨ıve Faster R-CNN. By this we mean that the RoI pooling layer is na¨ıvely adopted after the last convolutional layer (conv53 for VGG-16, inc5b for GoogleNet, and res5c for ResNet). In all cases, we set the output resolution of RoI pooling as 7×7. This is followed by a 81-d classifier (equivalent to a 1fc NoC).
Na¨更快的R-CNN。我们的意思是在最后一个卷积层之后才会采用RoI池化层（vGG-16为conv53，GoogleNet为inc5b，ResNet为res5c）。在所有情况下，我们将RoI池的输出分辨率设置为7×7。接下来是81-d分类器（相当于1fc NoC）。

Table 8 shows that VGG-16 has better AP (21.2%) than both GoogleNet (15.2%) and ResNet (16.9%), even though VGG-16 has worse image-level classification accuracy on ImageNet. One reason is that VGG-16 has a stride of 16 pixels on conv53, but GoogleNet and ResNet have a stride of 32 pixels on inc5b and res5c respectively. We hypothesize that a finer-resolution feature map (i.e., a smaller stride) contributes positively to object detection accuracy. To verify this, we reduce the stride of GoogleNet/ResNet from 32 to 16 by modifying the last stride=2 operation as stride=1. Then we adopt the "hole algorithm" [33], [34] ("Algorithme `a trous" [35]) on all following layers to compensate this modification. With a stride of 16 pixels, na¨ıve Faster R-CNN still performs unsatisfactorily, with an AP of 18.6% for GoogleNet and 21.3% for ResNet.
表8显示VGG-16具有比GoogleNet（15.2％）和ResNet（16.9％）更好的AP（21.2％），尽管VGG-16在ImageNet上具有更差的图像级分类准确度。一个原因是VGG-16在conv53上的步幅为16像素，但GoogleNet和ResNet在inc5b和res5c上的步幅分别为32像素。我们假设一个分辨率特征图（即一个较小的步幅）对物体检测精度有积极贡献。为了验证这一点，我们通过将最后一个stride = 2操作修改为stride = 1，将GoogleNet / ResNet的步幅从32减少到16。然后我们在所有后续层上采用"洞算法"[33]，[34]（"算法""[35]）来补偿这种修改。凭借16像素的步幅，更快的R-CNN仍然表现不尽如人意，GoogleNet的AP为18.6％，ResNet的AP为21.3％。

We argue that this is because in the case of na¨ıve Faster R-CNN, VGG-16 has a 3fc NoC but GoogleNet and ResNet has a 1fc NoC (Table 8). As we observed in the above section, a deeper region-wise NoC is important, even though GoogleNet and ResNet have deeper feature maps.
我们认为这是因为在更快的R-CNN的情况下，VGG-16具有3fc NoC但GoogleNet和ResNet具有1fc NoC（表8）。正如我们在上一节中所观察到的，即使GoogleNet和ResNet具有更深的特征映射，更深入的区域NoC也很重要。

Using MLP as NoC. Using the same settings of feature maps, we build a deeper MLP NoC by using 3 fc layers (f4096-f4096-fc81). As GoogleNet and ResNet have no pre-trained fc layers available, these layers are randomly initialized which we expect to perform reasonably (Sec. 3.4). This 3fc NoC significantly improves AP by about 4 to 5% for ResNet (21.3% to 26.3% with a stride of 16, and 16.9% to 21.2% with a stride of 32). These comparisons justify the importance of a deeper NoC.
使用MLP作为NoC。使用相同的特征映射设置，我们通过使用3个fc层（f4096-f4096-fc81）构建更深的MLP NoC。由于GoogleNet和ResNet没有预先训练好的fc层，因此这些层是随机初始化的，我们希望它们能够合理地执行（第3.4节）。这个3fc NoC显着提高了ResNet的AP约4％到5％（21.3％到26.3％，步幅为16,16.9％到21.2％，步幅为32）。这些比较证明了更深的NoC的重要性。

net	feature	stride	NoC	AP	AP@0.5	AP@0.75
VGG-16	$conv5_3$	16	fc_{4096}, fc_{4096}, fc_{81}	21.2	41.5	19.7
GoogleNet	inc5b	32	fc_{81}	15.2	34.7	11.6
GoogleNet	inc5b	32	fc_{4096}, fc_{4096}, fc_{81}	19.8	40.8	17.5
GoogleNet	inc5b, *à trous*	16	fc_{81}	18.6	39.4	15.8
GoogleNet	inc5b, *à trous*	16	fc_{4096}, fc_{4096}, fc_{81}	23.6	43.4	23.0
GoogleNet	inc4d	16	inc4e,5a,5b, fc_{81}	24.8	44.4	25.2
ResNet-101	res5c	32	fc_{81}	16.9	39.6	12.1
ResNet-101	res5c	32	fc_{4096}, fc_{4096}, fc_{81}	21.2	43.1	18.9
ResNet-101	res5c, *à trous*	16	fc_{81}	21.3	44.4	18.3
ResNet-101	res5c, *à trous*	16	fc_{4096}, fc_{4096}, fc_{81}	26.3	48.1	25.9
ResNet-101	$res4b_{22}$	16	res5a,5b,5c, fc_{81}	**27.2**	**48.4**	**27.6**

Table 8: Detection results of Faster R-CNN on the MS COCO val set. "inc" indicates an inception block, and "res" indicates a residual block.

表8：MS COCO val组上更快的R-CNN的检测结果。"inc"表示初始块，"res"表示残余块。

method	NoC	AP on COCO	mAP on VOC07
res5c, *à trous*	fc_{n+1}	21.3	71.9
res5c, *à trous*	fc_{4096}, fc_{4096}, fc_{n+1}	26.3	76.4
$res4b_{22}$	res5a,5b,5c, fc_{n+1}	**27.2**	**76.4**

Table 9: Detection results of Faster R-CNN + ResNet101 on MS COCO val (trained on MS COCO train) and PASCAL VOC 2007 test (trained on 07+12), based on different NoC structures.

表9：基于不同的NoC结构，在MS COCO val（在MS COCO列车上训练）和PASCAL VOC 2007测试（在07 + 12上训练）上更快的R-CNN + ResNet101的检测结果。

Using ConvNet as NoC. To build a convolutional NoC, we move the RoI pooling layer from the last feature map to an intermediate feature map that has a stride of 16 pixels (inc4d for GoogleNet and res4b22 for ResNet). The following convolutional layers (inc4e,5a,5b for GoogleNet and res5a,5b,5c for ResNet) construct the convolutional NoC. The `a trous trick is not necessary in this case.

使用ConvNet作为NoC。为了构建卷积NoC，我们将RoI池化层从最后一个特征映射移动到具有16像素步长的中间特征映射（GoogleNet为inc4d，ResNet为res4b22）。以下卷积层（GoogleNet的inc4e，5a，5b和ResNet的res5a，5b，5c）构成卷积NoC。在这种情况下，不需要一个琐碎的技巧。

With the deeper convolutional NoC, the AP is further improved, e.g., from 26.3% to 27.2% for ResNet. In particular, this NoC greatly improves localization accuracy — ResNet's AP@0.75 is increased by 1.7 points (from 25.9% to 27.6%) whereas AP@0.5 is nearly unchanged (from 48.1% to 48.4%). This observation is consistent with that on PASCAL VOC (Fig. 3), where a deep convolutional NoC improves localization.

随着更强的卷积NoC，AP进一步改善，例如ResNet的26.3％到27.2％。特别是，这个NoC大大提高了本地化的准确性 - ResNet的AP@0.75增加了1.7分（从25.9％增加到27.6％），而AP @0.5几乎没有变化（从48.1％增加到

48.4％）。该观察结果与PASCAL VOC（图3）的结果一致，深度卷积NoC改善了定位。

Table 9 shows the comparisons on PASCAL VOC for Faster R-CNN + ResNet-101. Both MLP and ConvNet as NoC (76.4%) perform considerably better than the 1fc NoC baseline (71.9%), though the benefit of using ConvNet as NoC is diminishing in this case.
表9显示了更快的R-CNN + ResNet-101对PASCAL VOC的比较。MLP和ConvNet作为NoC（76.4％）的表现明显优于1fc NoC基线（71.9％），尽管在这种情况下使用ConvNet作为NoC的好处正在减少。

Discussions The above system (27.2% AP and 48.4% AP@0.5) is the foundation of the detection system in the ResNet paper [17]. Combining with orthogonal improvements, the results in [17] secured the 1st place in MS COCO and ImageNet 2015 challenges.
讨论上述系统（27.2％AP和48.4％AP@0.5）是ResNet论文[17]中检测系统的基础。结合正交改进，[17]的结果在MS COCO和ImageNet 2015挑战中获得第一名。

The ablation results in Table 8 indicate that despite the effective Faster R-CNN and ResNet, it is not direct to achieve excellent object detection accuracy. In particular, a na¨ıve version of Faster R-CNN using ResNet has low accuracy (21.3% AP), because its regionwise classifier is shallow and not convolutional. On the contrary, a deep and convolutional NoC is an essential factor for Faster R-CNN + ResNet to perform accurate object detection.
表8中的消融结果表明，尽管有效的快速R-CNN和ResNet，但不能直接实现优异的物体检测精度。特别是，使用ResNet的更快的R-CNN版本具有低精度（21.3％AP），因为其区域分类是浅的而不是卷积的。相反，深度和卷积NoC是更快的R-CNN + ResNet执行精确物体检测的关键因素。

5 CONCLUSION
5结论

In this work, we delve into the detection systems and provide insights about the region-wise classifiers. We discover that deep convolutional classifiers are just as important as deep convolutional feature extractors. Based on the observations from the NoC perspective, we present a way of using Faster R-CNN with ResNets, which achieves nontrivial results on challenging datasets including MS COCO.
在这项工作中，我们深入研究了检测系统，并提供了有关区域分类的见解。我们发现深度卷积分类器与深度卷积特征提取器一样重要。基于NoC观点的观察结果，我们提出了一种使用更快的R-CNN和ResNets的方法，这种方法可以在包括MS COCO在内的具有挑战性的数据集上实现非常重要的结果。

REFERENCES

[1] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in CVPR, 2005.
[1] N. Dalal和B. Triggs，“人类检测的定向梯度直方图”，CVPR，2005年。

[2] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, “Object detection with discriminatively trained partbased models,” TPAMI, 2010.
[2] P. F. Felzenszwalb，R。B. Girshick，D。McAllester和D. Ramanan，“使用有区别训练的部分模型进行物体检测”，TPAMI，2010。

[3] X. Wang, M. Yang, S. Zhu, and Y. Lin, “Regionlets for generic object detection,” in ICCV, 2013.
[3] X. Wang，M。Yang，S。Zhu和Y. Lin，“通用物体检测区域小组”，ICCV，2013年。

[4] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders, “Selective search for object recognition,” IJCV, 2013.
[4] J. R. Uijlings，K。E. van de Sande，T。Gevers和A. W. Smeulders，“选择性搜索物体识别”，IJCV，2013。

[5] A. Krizhevsky, I. Sutskever, and G. Hinton, “Imagenet classification with deep convolutional neural networks,” in NIPS, 2012.
[5] A. Krizhevsky，I。Sutskever和G. Hinton，“深度卷积神经网络的Imagenet分类”，NIPS，2012年。

[6] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in CVPR, 2014.
[6] R. Girshick，J。Donahue，T。Darrell和J. Malik，“用于精确对象检测和语义分割的丰富特征层次”，在CVPR，2014中。

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. FeiFei, “Imagenet: A large-scale hierarchical image database,” in CVPR, 2009.
[7] J. Deng，W。Dong，R。Socher，L.-J。 Li，K。Li和L. FeiFei，“Imagenet：一个大规模的分层图像数据库”，在CVPR，2009年。

[8] P.-A. Savalle, S. Tsogkas, G. Papandreou, and I. Kokkinos, “Deformable part models with CNN features,” in Parts and Attributes Workshop, ECCV, 2014.
[8] P.-A. Savalle，S。Tsogkas，G。Papandreou和I. Kokkinos，“具有CNN功能的可变形零件模型”，部件和属性研讨会，ECCV，2014年。

[9] R. Girshick, F. Iandola, T. Darrell, and J. Malik, “Deformable part models are convolutional neural networks,” in CVPR, 2015.
[9] R. Girshick，F。Iandola，T。Darrell和J. Malik，“可变形零件模型是卷积神经网络”，在CVPR，2015年。

[10] L. Wan, D. Eigen, and R. Fergus, “End-to-end integration of a convolutional network, deformable parts model and nonmaximum suppression,” in CVPR, 2015.
[10] L. Wan，D。Eigen和R. Fergus，“卷积网络的端到端集成，可变形零件模型和非最大抑制”，在CVPR，2015年。

[11] W. Y. Zou, X. Wang, M. Sun, and Y. Lin, “Generic object detection with dense neural patterns and regionlets,” in BMVC, 2014.
[11] W. Y. Zou，X。Wang，M。Sun和Y. Lin，“密集神经模式和区域小组的通用物体检测”，BMVC，2014年。

[12] K. He, X. Zhang, S. Ren, and J. Sun, “Spatial pyramid pooling in deep convolutional networks for visual recognition,” in ECCV, 2014.
[12] K. He，X。Zhang，S。Ren和J. Sun，“用于视觉识别的深度卷积网络中的空间金字塔汇集”，ECCV，2014年。

[13] R. Girshick, “Fast R-CNN,” in ICCV, 2015.
[13] R. Girshick，“快速R-CNN”，ICCV，2015年。

[14] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” in NIPS, 2015.
[14] S. Ren，K。He，R。Girshick和J. Sun，“更快的R-CNN：与区域提案网络进行实时对象检测”，NIPS，2015年。

[15] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in ICLR, 2015.
[15] K. Simonyan和A. Zisserman，“用于大规模图像识别的非常深的卷积网络”，载于ICLR，2015年。

[16] I. J. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio, “Maxout networks,” arXiv:1302.4389, 2013.
[16] I. J. Goodfellow，D。Warde-Farley，M。Mirza，A。Courville和Y. Bengio，“Maxout networks，”arXiv：1302.4389,2013。

[17] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in arXiv prepring arXiv:1506.01497, 2015.
[17] K. He，X。Zhang，S。Ren和J. Sun，“图像识别的深度残留学习”，arXiv prepring arXiv：1506.01497,2015。

[18] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, and A. Rabinovich, “Going deeper with convolutions,” Tech. Rep., 2014. [Online]. Available: http://arxiv.org/pdf/1409.4842v1
[18] C. Szegedy，W。Liu，Y。Jia，P。Sermanet，S。Reed，D。Anguelov，D。Erhan和A. Rabinovich，“深入研究”，Tech。 Rep。 ，2014。[在线]。可用：http：//arxiv.org/pdf/1409.4842v1

[19] P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” in CVPR, 2001.
[19] P. Viola和M. Jones，“快速物体检测使用增强级联的简单特征”，在CVPR，2001年。

[20] S. Lazebnik, C. Schmid, and J. Ponce, “Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories,” in CVPR, 2006.
[20] S. Lazebnik，C。Schmid和J. Ponce，“超越特征袋：用于识别自然场景类别的空间金字塔匹配”，CVPR，2006年。

[21] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” IJCV, 2004.
[21] D. G. Lowe，“尺度不变关键点的独特图像特征”，IJCV，2004。

[22] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, “Overfeat: Integrated recognition, localization and detection using convolutional networks,” in ICLR, 2014.
[22] P. Sermanet，D。Eigen，X。Zhang，M。Mathieu，R。Fergus和Y. LeCun，“Overfeat：使用卷积网络的综合识别，定位和检测”，ICLR，2014年。

[23] K. Lenc and A. Vedaldi, “R-cnn minus r,” in BMVC, 2015.
[23] K. Lenc和A. Vedaldi，“R-cnn减去r”，BMVC，2015年。

[24] S. Gidaris and N. Komodakis, “Object detection via a multiregion & semantic segmentation-aware cnn model,” in ICCV, 2015.
[24] S. Gidaris和N. Komodakis，“通过多区域和语义分割感知的cnn模型进行对象检测”，ICCV，2015年。

[25] M. D. Zeiler and R. Fergus, “Visualizing and understanding convolutional neural networks,” in ECCV, 2014.
[25] M. D. Zeiler和R. Fergus，“可视化和理解卷积神经网络”，在ECCV，2014年。

[26] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, “The PASCAL Visual Object Classes (VOC) Challenge,” IJCV, 2010.
[26] M. Everingham，L。Van Gool，C。K. Williams，J。Winn和A. Zisserman，“PASCAL视觉对象类（VOC）挑战”，IJCV，2010。

[27] P. Agrawal, R. Girshick, and J. Malik, “Analyzing the performance of multilayer neural networks for object recognition,” in ECCV, 2014.
[27] P. Agrawal，R。Girshick和J. Malik，“分析用于物体识别的多层神经网络的性能”，载于ECCV，2014年。

[28] K. Hornik, M. Stinchcombe, and H. White, “Multilayer feedforward networks are universal approximators,” Neural networks, 1989.
[28] K. Hornik，M。Stinchcombe和H. White，“多层前馈网络是通用逼近器”，神经网络，1989。

[29] V. Nair and G. E. Hinton, “Rectified linear units improve restricted boltzmann machines,” in ICML, 2010.
[29] V. Nair和G. E. Hinton，“Recti fi ed linear units改进限制的boltzmann机器”，ICML，2010。

[30] D. Hoiem, Y. Chodpathumwan, and Q. Dai, “Diagnosing error in object detectors,” in ECCV, 2012.
[30] D. Hoiem，Y。Chodpathumwan和Q.Dai，“诊断物体探测器中的误差”，ECCV，2012年。

[31] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Doll´ar, and C. L. Zitnick, “Microsoft coco: Common objects in context,” arXiv:1405.0312, 2014.
[31] T.-Y. Lin，M。Maire，S。Belongie，J。Hays，P。Perona，D。Ramanan，P。Doll'ar和C. L. Zitnick，“Microsoft coco：Common objects in context，”arXiv：1405.0312,2014。

[32] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” in ICML, 2015.
[32] S. Ioffe和C. Szegedy，“批量标准化：通过减少内部协变量变化来加速深层网络培训”，ICML，2015年。

[33] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in CVPR, 2015.
[33] J. Long，E。Shelhamer和T. Darrell，“用于语义分割的完全卷积网络”，在CVPR，2015年。

[34] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, “Semantic image segmentation with deep convolutional nets and fully connected crfs,” in ICLR, 2015.
[34] L.-C. Chen，G。Papandreou，I。Kokkinos，K。Murphy和A. L. Yuille，“深度卷积网和完全连接的crfs的语义图像分割”，ICLR，2015年。

[35] S. Mallat, A wavelet tour of signal processing. Academic press, 1999.
[35] S. Mallat，信号处理的小波导览。学术出版社，1999年。